

# 软件测试—白盒测试 判定测试

---

## （一）基本思想

对源代码中的所有逻辑值都测试真值与假值的情况

## （二）几种测试覆盖指标

1. 语句覆盖
2. 条件覆盖
3. 判定覆盖
4. 判定条件覆盖
5. 条件组合覆盖
6. 修正判定/条件覆盖

### （1）语句覆盖

**基本思想**

设计测试用例应保证程序的**每一条可执行语句**至少执行一次，无需保证每个判定分支都要执行，若一个测试用例可以覆盖所有的语句，则设计一个即可

### （2）判定覆盖

**基本思想**

每一个**判定节点**都要进行测试其取真，取假的情况，同样的，在此中覆盖中也无需保证每一个小分支都要被测试到

### （3）条件覆盖

**基本思想**

每一个**简单判定条件**都要测试其取真与取假的情况

### （4）判定条件覆盖

**基本思想**

设计测试用例时，**既要满足每个判定节点都要进行取真与取假的测试，每一个简单判定条件也需要检验其取真与取假的情况**

### （5）条件组合覆盖（真值表）

**基本思想**

设计测试用例应满足，在每个判定节点中，**所有的简单判定条件所有可能的取值集合**情况都被执行一次

### （6）修正的判定条件覆盖（传说考的概率不大）

**基本思想**

在满足判定/条件覆盖的基础上，每一个单独的判定条件**都能独立的影响**整个判定表达式的取值

**设计用例时的注意事项**

1. 避免测试数据受到符合判定表达式的“与”，“或”关系的屏蔽效应
2. 尽量结合边界值选择数据